

```
/* Query 1:
```

10

15

20

25

30

35

40

45

50

55

```

eq_ii->x1      = (ePtr)5;
eq_ii->x2      = (ePtr)6;

/* s.#1 */
5  thread[5].opc = eVARATT;
   thread[5].typ = eT_int;
   thread[5].obj = (ePtr)offsetof(varatt3, heap);
   varatt3->tvar = (ePtr)9;
   varatt3->col  = 1;
10
   /* t.#0 */
   thread[6].opc = eVARATT;
   thread[6].typ = eT_int;
   thread[6].obj = (ePtr)offsetof(varatt4, heap);
15  varatt4->tvar = (ePtr)10;
   varatt4->col  = 0;

   /* TA S s */
   thread[7].opc = eTA;
20  thread[7].typ = eT_streamof(c10intsml);
   thread[7].obj = (ePtr)offsetof(ta1, heap);
   ta1->tvar      = (ePtr)9;
   ta1->pred      = (ePtr)0;
   ta1->tmd       = t->pTMD;
25  ta1->openp    = 0;

   /* TA T t */
   thread[8].opc = eTA;
   thread[8].typ = eT_streamof(intvc10);
30  thread[8].obj = (ePtr)offsetof(ta2, heap);
   ta2->tvar      = (ePtr)10;
   ta2->pred      = (ePtr)0;
   ta2->tmd       = t1->pTMD;
   ta2->openp    = 0;
35

   /* VAR s */
   thread[9].opc = eVAR;
   thread[9].typ = (eType)c10intsml;

40  /* VAR t */
   thread[10].opc = eVAR;
   thread[10].typ = (eType)intvc10;

   /* EXIT */
45  thread[11].opc = eEXIT;
   thread[11].typ = 0;
#endif

/* Query 2:
50
   select  x, b
     from  S, T
    where  y = a

55  Plan:

      BNO -- [ s.#0, t. #1 ]

```

```

      |
      |
      | NLJ -- [ s.#1 == t.# 0 ]
      / \
5     /   \
    TA S s   ITA T t -- [ t.#0 > s.#1 ]
                                stop
    */

10  #if 0
    /* BNO */
    thread[0].opc = eBNO;
    thread[0].typ = eT_streamof(c10vc10);
    thread[0].obj = (ePtr)objoffset(bno, heap);
15  bno->in      = (ePtr)3;
    bno->fn      = (ePtr)1;
    bno->fns     = 2;
    bno->bno     = oMemAlloc(bno->fns * sizeof(ColVal));

20  /* s.#0 */
    thread[1].opc = eVARATT;
    thread[1].typ = eT_setlen(eT_char, 10);
    thread[1].obj = (ePtr)objoffset(varatt1, heap);
    varatt1->tvar = (ePtr)9;
25  varatt1->col  = 0;

    /* t.#1 */
    thread[2].opc = eVARATT;
    thread[2].typ = eT_setlen(eT_vchar, 10);
30  thread[2].obj = (ePtr)objoffset(varatt2, heap);
    varatt2->tvar = (ePtr)10;
    varatt2->col  = 1;

    /* NLJ */
35  thread[3].opc = eNLJ;
    thread[3].typ = 0;
    thread[3].obj = (ePtr)objoffset(nlj, heap);
    nlj->outer    = (ePtr)7;
    nlj->inner    = (ePtr)8;
40  nlj->pred     = (ePtr)11;

    /* < */
    thread[4].opc = eLT_II;
    thread[4].typ = eT_bool;
45  thread[4].obj = (ePtr)objoffset(lt_ii, heap);
    lt_ii->x1     = (ePtr)6;
    lt_ii->x2     = (ePtr)5;

    /* t.#0 */
50  thread[5].opc = eVARATT;
    thread[5].typ = eT_int;
    thread[5].obj = (ePtr)objoffset(varatt4, heap);
    varatt4->tvar = (ePtr)10;
    varatt4->col  = 0;

55  /* s.#1 */
    thread[6].opc = eVARATT;

```

5

10

15

20

30

35

40

45

50

55

SORT S

```

      |
      |
      | TA S
    */
5
#if 0
    /* BNO * */
    thread[0].opc = eBNO;
    thread[0].typ = (eType)c10intsml;
10    thread[0].obj = (ePtr)objoffset(bno, heap);
    bno->in      = (ePtr)1;
    bno->fn       = (ePtr)0;
    bno->fns      = 3;
    bno->bno      = oMemAlloc(bno->fns * sizeof(ColVal));
15
    /* SORT s */
    thread[1].opc = eSORT;
    thread[1].typ = eT_streamof(c10intsml);
    thread[1].obj = (ePtr)objoffset(sort, heap);
20    sort->in      = (ePtr)2;
    sort->tvar     = (ePtr)0;
    sort->tmd      = query->resultset.pTMD;
    oStrcpy(sort->tmd->tname, "sorting");
    sort->sortp    = 0;
25
    /* TA S */
    thread[2].opc = eTA;
    thread[2].typ = eT_streamof(c10intsml);
    thread[2].obj = (ePtr)objoffset(tal, heap);
30    tal->tvar     = (ePtr)0;
    tal->pred      = (ePtr)0;
    tal->tmd       = t->pTMD;
    tal->openp     = 0;
35
    /* EXIT */
    thread[3].opc = eEXIT;
    thread[3].typ = 0;
#endif
40
    /* Query 4:

        select  sum(y),count(*)
          from  S

45    Plan:

        BNOAGG -- [ SUM_I,  CNT ]
          |           |
          |           | s.#1
          |           |
50    TA S s

    */

#if 0
    /* BNOAGG */
55    thread[0].opc = eBNOAGG;
    thread[0].typ = (eType)intint;
    thread[0].obj = (ePtr)objoffset(bno, heap);

```

10050063-030102

```

bno->in      = (ePtr)4;
bno->fn      = (ePtr)1;
bno->fns     = 2;
bno->bno     = oMemAlloc(bno->fns * sizeof(ColVal));
5
/* SUM_I */
thread[1].opc = eSUM_I;
thread[1].typ = eT_int;
thread[1].obj = (ePtr)objoffset(sum, heap);
10
sum->in      = (ePtr)3;

/* CNT */
thread[2].opc = eCNT;
thread[2].typ = eT_int;
15

/* s.#1 */
thread[3].opc = eVARATT;
thread[3].typ = eT_int;
thread[3].obj = (ePtr)objoffset(varatt1, heap);
20
varatt1->tvar = (ePtr)5;
varatt1->col  = 1;

/* TA S */
thread[4].opc = eTA;
thread[4].typ = eT_streamof(c10intsml);
25
thread[4].obj = (ePtr)objoffset(ta1, heap);
ta1->tvar      = (ePtr)5;
ta1->pred      = (ePtr)0;
ta1->tmd       = t->pTMD;
30
ta1->openp     = 0;

/* VAR s */
thread[5].opc = eVAR;
thread[5].typ = (eType)c10intsml;
35

/* EXIT */
thread[5].opc = eEXIT;
thread[5].typ = 0;
#endif
40

/* Query 5:

select  (select (sum(a)
              from T
45              where a > 1)
        from S

Plan:

50      BNO -- ATT #0
        |
        | SUBQ
        |
        | AGG -- SUM_I
55      |   |   |   |
        |   |   |   | t.#0
        |   |   |   |
        TA S   |   |   |

```

```

                                TA T t -- [ t.#0 > 1 ]
        */

    #if 0
5      /* BNO */
        thread[0].opc = eBNO;
        thread[0].typ = (eType)xint;
        thread[0].obj = (ePtr)offsetof(bno, heap);
10     bno->in      = (ePtr)2;
        bno->fn      = (ePtr)1;
        bno->fns     = 1;
        bno->bno     = oMemAlloc(bno->fns * sizeof(ColVal));

        /* ATT #0 */
15     thread[1].opc = eATT;
        thread[1].typ = eT_int;
        thread[1].obj = (ePtr)offsetof(att, heap);
        att->in      = (ePtr)3;
        att->col     = 0;

20     /* TA S */
        thread[2].opc = eTA;
        thread[2].typ = eT_streamof(c10intsm1);
        thread[2].obj = (ePtr)offsetof(ta1, heap);
25     ta1->tvar     = (ePtr)0;
        ta1->pred    = (ePtr)0;
        ta1->tmd     = t->pTMD;
        ta1->openp   = 0;

30     /* SUBQ */
        thread[3].opc = eSUBQ;
        thread[3].typ = eT_streamof(xint);
        thread[3].obj = (ePtr)offsetof(subq, heap);
35     subq->in      = (ePtr)4;
        subq->tvar    = (ePtr)0;

        /* AGG */
        thread[4].opc = eAGG;
        thread[4].typ = eT_streamof(xint);
40     thread[4].obj = (ePtr)offsetof(agg, heap);
        agg->in      = (ePtr)5;
        agg->fn      = (ePtr)6;
        agg->fns     = 1;

45     /* TA T t */
        thread[5].opc = eTA;
        thread[5].typ = eT_streamof(intvc10);
        thread[5].obj = (ePtr)offsetof(ta2, heap);
        ta2->tvar     = (ePtr)10;
50     ta2->pred     = (ePtr)8;
        ta2->tmd     = t1->pTMD;
        ta2->openp   = 0;

        /* SUM_I */
55     thread[6].opc = eSUM_I;
        thread[6].typ = eT_int;
        thread[6].obj = (ePtr)offsetof(sum, heap);

```

100500068-030102

```

sum->in      = (ePtr)7;

/* t.#0 */
thread[7].opc = eVARATT;
thread[7].typ = eT_int;
thread[7].obj = (ePtr)objoffset(varatt1, heap);
varatt1->tvar = (ePtr)10;
varatt1->col  = 0;

/* < */
thread[8].opc = eLT_II;
thread[8].typ = eT_bool;
thread[8].obj = (ePtr)objoffset(lt_ii, heap);
lt_ii->x1      = (ePtr)9;
lt_ii->x2      = (ePtr)7;

/* 1 */
thread[9].opc = eLIT;
thread[9].typ = eT_int;
eT_as_int(thread[9].val) = 1;

/* VAR t */
thread[10].opc = eVAR;
thread[10].typ = (eType)intvc10;

/* EXIT */
thread[11].opc = eEXIT;
thread[11].typ = 0;
#endif

/* Query 6:

select  count(z),x,y
  from  S
group by  x,y

Plan:

      BNO *
      |
      | GROUP +-- [ CNT ]
      |      |
      |      +-- [ MOV, VARATT s.#1 ]
      |      |
      |      SORT as s  VARATT s.#0
      |      |
      |      TA S
      |
      */

#if 0
/* BNO * */
thread[0].opc = eBNO;
thread[0].typ = (eType)intc10int;

```



```

thread[0].obj = (ePtr)objoffset(bno, heap);
bno->in      = (ePtr)1;
bno->fn      = (ePtr)0;
bno->fns     = 3;
5  bno->bno    = oMemAlloc(bno->fns * sizeof(ColVal));

/* GROUP */
thread[1].opc = eGROUP;
thread[1].typ = eT_streamof(c10intint);
10 thread[1].obj = (ePtr)objoffset(group, heap);
group->in     = (ePtr)5;
group->src    = (ePtr)5;
group->agg    = (ePtr)2;
group->aggs   = 1;
15 group->fns  = 2;

/* CNT */
thread[2].opc = eCNT;
thread[2].typ = eT_int;
20

/* MOV */
thread[3].opc = eMOV;
thread[3].typ = eT_setlen(eT_char, 10);
thread[3].obj = (ePtr)objoffset(mov, heap);
25 mov->in     = (ePtr)8;
mov->n       = 10+1;

/* VARATT s.#1 */
thread[4].opc = eVARATT;
thread[4].typ = eT_int;
30 thread[4].obj = (ePtr)objoffset(varatt2, heap);
varatt2->tvar = (ePtr)7;
varatt2->col  = 1;

/* SORT */
thread[5].opc = eSORT;
thread[5].typ = eT_streamof(c10intsml);
thread[5].obj = (ePtr)objoffset(sort, heap);
35 sort->in     = (ePtr)6;
sort->tvar    = (ePtr)7;
sort->tmd     = gb->table->pTMD;
oStrcpy(sort->tmd->tname, "teggyl1");
40 sort->sortp  = 0;

/* TA S */
thread[6].opc = eTA;
thread[6].typ = eT_streamof(c10intsml);
thread[6].obj = (ePtr)objoffset(ta1, heap);
45 ta1->tvar    = (ePtr)0;
ta1->pred     = (ePtr)0;
50 ta1->tmd     = t->pTMD;
ta1->openp    = 0;

/* VAR s */
55 thread[7].opc = eVAR;
thread[7].typ = (eType)c10intsml;

```

```

/* VARATT s.#0 */
thread[8].opc = eVARATT;
thread[8].typ = eT_setlen(eT_char, 10);
thread[8].obj = (ePtr)offsetof(varatt1, heap);
5  varatt1->tvar = (ePtr)7;
   varatt1->col = 0;

/* EXIT */
thread[9].opc = eEXIT;
10  thread[9].typ = 0;
#endif

#if 0
15  /* BNO * */
   thread[0].opc = eBNO;
   thread[0].typ = (eType)intd105int;
   thread[0].obj = (ePtr)offsetof(bno, heap);
   bno->in      = (ePtr)1;
20  bno->fn      = (ePtr)0;
   bno->fns     = 3;
   bno->bno     = oMemAlloc(bno->fns * sizeof(ColVal));

/* GROUP */
25  thread[1].opc = eGROUP;
   thread[1].typ = eT_streamof(d105intint);
   thread[1].obj = (ePtr)offsetof(group, heap);
   group->in     = (ePtr)5;
   group->src     = (ePtr)5;
30  group->agg    = (ePtr)2;
   group->aggs   = 1;
   group->fns    = 2;

/* CNT */
35  thread[2].opc = eCNT;
   thread[2].typ = eT_int;

/* VARATT s.#0 */
thread[3].opc = eMOV;
40  thread[3].typ = eT_setscale(eT_setprec(eT_decimal, 10), 5);
   thread[3].obj = (ePtr)offsetof(mov, heap);
   mov->in       = (ePtr)8;
   mov->n        = MAX_INTERNAL_DECIMAL_BYTES;

45  /* VARATT s.#1 */
   thread[4].opc = eVARATT;
   thread[4].typ = eT_int;
   thread[4].obj = (ePtr)offsetof(varatt2, heap);
   varatt2->tvar = (ePtr)7;
50  varatt2->col = 1;

/* SORT */
thread[5].opc = eSORT;
thread[5].typ = eT_streamof(d105intsml);
55  thread[5].obj = (ePtr)offsetof(sort, heap);
   sort->in      = (ePtr)6;
   sort->tvar     = (ePtr)7;

```

```

sort->tmd      = gb->table->pTMD;
oStrcpy(sort->tmd->tname, "teggyl0");
sort->sortp    = 0;

5   /* TA S */
thread[6].opc = eTA;
thread[6].typ = eT_streamof(d105intsml);
thread[6].obj = (ePtr)objoffset(ta1, heap);
ta1->tvar      = (ePtr)0;
10  ta1->pred   = (ePtr)0;
ta1->tmd       = t->pTMD;
ta1->openp    = 0;

/* VAR s */
15  thread[7].opc = eVAR;
thread[7].typ = (eType)d105intsml;

/* VARATT s.#0 */
thread[8].opc = eVARATT;
20  thread[8].typ = eT_setscale(eT_setprec(eT_decimal, 10), 5);
thread[8].obj = (ePtr)objoffset(varatt1, heap);
varatt1->tvar = (ePtr)7;
varatt1->col  = 0;

25  /* EXIT */
thread[9].opc = eEXIT;
thread[9].typ = 0;
#endif

30  /* Query 7:

select  x + y
from    S

Plan:

      BNO ----- [ ADD_DI ]
      |             /      \
      |            /        \
40   TA S s  VARATT s.#1  VARATT s.#2
      */

55  #if 0
/* BNO */
thread[0].opc = eBNO;
thread[0].typ = (eType)d105;
thread[0].obj = (ePtr)objoffset(bno, heap);
bno->in       = (ePtr)4;
50  bno->fn     = (ePtr)1;
bno->fns      = 1;
bno->bno      = oMemAlloc(bno->fns * sizeof(ColVal));

/* ADD_DI */
thread[1].opc = eADD_DI;
thread[1].typ = eT_setscale(eT_setprec(eT_decimal, 10), 5);
thread[1].obj = (ePtr)objoffset(add_di, heap);

```

10090063-030102

```

add_di->x1    = (ePtr)2;
add_di->x2    = (ePtr)3;

```

```

/* VARATT */

```

```

5  thread[2].opc = eVARATT;
   thread[2].typ = eT_setscale(eT_setprec(eT_decimal, 10), 5);
   thread[2].obj = (ePtr)offsetof(varatt1, heap);
   varatt1->tvar = (ePtr)5;
   varatt1->col  = 1;

```

```

/* VARATT */

```

```

10 thread[3].opc = eVARATT;
   thread[3].typ = eT_int;
   thread[3].obj = (ePtr)offsetof(varatt2, heap);
15 varatt2->tvar = (ePtr)5;
   varatt2->col  = 2;

```

```

/* TA S */

```

```

20 thread[4].opc = eTA;
   thread[4].typ = eT_streamof(intd105int);
   thread[4].obj = (ePtr)offsetof(ta1, heap);
   ta1->tvar      = (ePtr)5;
   ta1->pred      = (ePtr)0;
   ta1->tmd       = t->pTMD;
25 ta1->openp     = 0;

```

```

/* VAR s */

```

```

30 thread[5].opc = eVAR;
   thread[5].typ = (eType)intd105int;

```

```

/* EXIT */

```

```

thread[6].opc = eEXIT;
thread[6].typ = 0;

```

```

35 #endif

```

```

/* Query 8:

```

```

   select  y - z
   from    S

```

```

   Plan:

```

```

      BNO ----- [ SUB_II ]

```

```

      |           /      \
      |          /        \
      |         /          \
45  TA S s  VARATT s.#1  VARATT s.#2

```

```

*/

```

```

/* BNO */

```

```

50 thread[0].opc = eBNO;
   thread[0].typ = (eType)xint;
   thread[0].obj = (ePtr)offsetof(bno, heap);
55 bno->in        = (ePtr)4;
   bno->fn        = (ePtr)1;
   bno->fns       = 1;

```

20100903 03:01:02

```

bno->bno      = oMemAlloc(bno->fns * sizeof(ColVal));

/* SUB_II */
thread[1].opc = eSUB_II;
thread[1].typ = eT_int;
thread[1].obj = (ePtr)objoffset(sub_ii, heap);
sub_ii->x1    = (ePtr)2;
sub_ii->x2    = (ePtr)3;

/* VARATT */
thread[2].opc = eVARATT;
thread[2].typ = eT_int;
thread[2].obj = (ePtr)objoffset(varatt1, heap);
varatt1->tvar = (ePtr)5;
varatt1->col  = 1;

/* VARATT */
thread[3].opc = eVARATT;
thread[3].typ = eT_smlint;
thread[3].obj = (ePtr)objoffset(varatt2, heap);
varatt2->tvar = (ePtr)5;
varatt2->col  = 2;

/* TA S */
thread[4].opc = eTA;
thread[4].typ = eT_streamof(c10intsml);
thread[4].obj = (ePtr)objoffset(ta1, heap);
ta1->tvar      = (ePtr)5;
ta1->pred      = (ePtr)0;
ta1->tmd       = t->pTMD;
ta1->openp     = 0;

/* VAR s */
thread[5].opc = eVAR;
thread[5].typ = (eType)c10intsml;

/* EXIT */
thread[6].opc = eEXIT;
thread[6].typ = 0;

ePrepare(&s);

while ((rc = eInterpret(&s)) == eOK) {
    eRender(&s, hostio);
    n++;
}

eSweep(&s);

oMemFreeSafe(bno->bno);
}

```